

# A SPREADSHEET APPLICATION FOR THE PLANNING AND TRACKING OF PROJECT LABOR HOURS AND EXPENDITURES

Logožar R.<sup>1</sup>, Havaš L.<sup>1</sup>

<sup>1</sup>Polytechnic of Varaždin, J. Križanića 33, HR-42000 Varaždin, Croatia

**Abstract.** The paper briefly outlines a comprehensive spreadsheet application for the planning and tracking of project labor hours and costs, including the material and outsourcing expenditures. Some specifics of the use of table-calculators for this sort of problems are depicted. The application is aimed for the projects in the international business environment, with many variable parameters. It consists of the versatile and easily expandable tables with selectable data form entry cells (e.g. the cost price can be entered in a selected currency). Where applied, this sort of input leads to the explicit separation of the entry cells from the calculation (presentation) cells, and improves the overall consistency of the tables. The spreadsheets give a wide range of project information, from the subtle details up to the general and summarizing facts. The conditional color-formatting of the cells is consistently used for better visualization of the data. The application is designed to serve a manager in planning, running, supervising and concluding the project.

**Key words:** project planning, project tracking, spreadsheet application, selectable data form entry cells.

**Sažetak.** Članak ukratko opisuje sveobuhvatnu aplikaciju izvedenu u tabličnom kalkulatoru za planiranje i praćenje troškova projekata, uključujući radne sate, te troškove materijala i vanjskih usluga. Navedene su neke specifičnosti uporabe tabličnih kalkulatora za ovu vrstu problema. Aplikacija je namijenjena za projekte u internacionalnom okruženju s mnogo varijabilnih parametara. Sastoji se od svestranih, lako proširivih tablica, s podatkovnim ćelijama s odaberivom formom unosa (npr. unos troškova u izabranoj valuti). Gdje je primijenjena, ovakva vrsta unosa dovodi do odvajanja ulaznih i prezentacijskih ćelija, uz poboljšanje konzistentnosti tablica. Tablična aplikacija daje široki raspon informacija o projektu, od finih detalja do općenitih i sumarnih činjenica. Uporabom uvjetno formatiranih ćelija ostvaren je pregledan prikaz podataka. Aplikacija je oblikovana da služi upravitelju za planiranje, vođenje, nadzor i zaključenje projekta.

**Ključne riječi:** planiranje projekta, praćenje projekta, aplikacija u tabličnom kalkulatoru, ulazne ćelije s odaberivom podatkovnom formom.

## 1. INTRODUCTION

Today we find a lot of solutions for project managing, among which the Microsoft® Project Manager (MSP) seems to be the best known commercial product [1]. It is a comprehensive software tool designed for developing

project plans, tracking the project progress, managing budgets and analyzing workloads. This and similar project managing programs are based on the concept of Gantt charts [2, 3], which offer the analysis of the critical path schedules and critical chains. Several of these are available as online applications, and many of them are advertised as free [4].

In this work we present a project managing application that takes a different approach. It puts the emphasis on the project's labor and other financial indicators. It all started with a "simple idea" of creating a table that would summarize the workers' activities and track their labor hours and earnings in one of our complex projects. In our case the table was started in the famous MS® Excel (e.g. see [5]). When creating the table, the basic functionality was achieved relatively easy and fast. But as it often happens in programming, a comprehensive solution required full restructuring of the initial concept of a "simple table" into a more elaborate data storage and manipulation system. Continue

During the application development, a few "custom" features urged us to pursue the initial different approach. That finally drove us to a point when giving up would mean throwing away many weeks of work. The additional motivation was that the ready-made solutions either didn't seem to answer our needs, or were not practical enough. When considering the use of MS® Project Manager to do that, we came to the conclusion that our special demands of many financial aspects, e.g. like the use of two currencies, different contracting models, tracking of the worker payments, etc. were not met, without even mentioning the MSP's high commercial price. So not only that we went on with the spreadsheets construction, but were making them more and more general in every development step. New features were added, until the tables provided a comprehensive tool for project planning and tracking. Some of the aspects, e.g. like the temporal ones, are not solved in as detailed and visually appealing way as in the above cited programs. Nevertheless, the timing of the jobs and work terms is presented with the relevant data and emphasized with the appropriate color formatting, and could be improved in the future.

Eventually, our Excel workbook grew from the "one-big-table" stage up to the 6-spreadsheet application, containing around a dozen of the crucial tables each consisting of 20 – 30 columns, and about 40 more helping tables for the side calculations, constant definitions, different data presentations, etc. Hoping that the application will help project planners and managers to project their visions and ideas into reality, we named it **Projector**.

Before getting into the application more closely, in the next section we expose a few general remarks on the relationship between the spreadsheet tables on one hand and databases on the other, together with a few specific features of the former.

## 2. THE SPREADSHEETS

The table calculators are widely popular among the broad range of users for their simplicity and versatility. They are intuitive and similar to the paper-based tables, with the benefit that the effects of data input are “instantly” visible on the computer screen, or can be made visible by simple scrolling or by choosing other worksheet. So, there are no menus and submenus to investigate, just relatively low-level table presentations. In our opinion, this *low level presentation* contributes to the user perception of being directly involved in the whole process: from the data collection and input, to the data analysis and interpretation. An *involved user* can be expected to get better results and be more satisfied with a computer application, even if the program functionality may be elementary and its appearance Spartan.

Another benefit is that the users acquainted with the basics of spreadsheet calculators are likely to accept a new such application as just “another table”. They expect it to be easy to fill in, and that the results will be somewhere there, hopefully presented in a good fashion.

From the developer’s standpoint, a well founded and structured spreadsheet tables are easy to modify and upgrade. One of the most demanding things in programming today—the user interface—is already there, together with many other tools. A knowledgeable user is tempted to tailor the tables and do the upgrades herself (or himself). Or this can be ordered from developers expecting much lower prices than for the full fledged applications. The simple changes, like renaming column titles, expanding and deleting table rows (without interfering the attribute scheme), hiding rows and columns, adapting data presentation tables, and similar, should be straightforward and easily accomplishable by an average user.

Many people regard the spreadsheet calculators irreplaceable, and find their own tables, tailored to their specific problems, as the most valuable assets in data collection and data analysis of their particular problems.

All these reasons justify the use of spreadsheets even for the demanding and complex applications. They could and should be comparable to those developed within database developing environments or as special programs. If not truly equal in all the functionality, they should at least be equal in presentation of the quantitative indicators, if not suitable for the big size databases, they are an excellent choice for the information extraction from data collected in an experiment or statistical research.

### 2.1 Spreadsheets and Databases

A condition sine qua non for all databases is to build them according to the Codd’s 13 rules and to keep them in the normal forms—preferably in the famous 3<sup>rd</sup> normal form. This applies to the databases written on paper, big relational databases that are run under expensive proprietary database engines, and for our project-managing

Excel tables. Every deviation from the formal Codd’s rules will be paid with the loss of data integrity, loss of correct referencing, the inability of consistent updating, deleting, etc. Thus, if a spreadsheet programmer does make a mistake in the formation of the tables, she (or he) will soon notice the introduced anomalies when trying to search, alter or update the data base. Up to now, we may say that for the initial, *data input, columns*, the general database rules must be followed.

As a curiosity, let’s mention that the revolutionary Codd’s paper also dealt with the project managing [6].

### 2.2 Dataflow – the constant recalculating

The very definition of spreadsheets is about calculating the new, higher level, information from the values entered into the table cells. The table cell can be considered as a variable. The restriction is that it can be accessed only through the corresponding cell, via manual data input. A special kind of the cell value is a formula entered into it, through which the cell can depend on other variables (cells). From the database standpoint, the introduction of such cells produces data redundancy. However, as we shall soon show, this redundancy is of a special kind, which is *data-wise consistent*. It is the spreadsheet’s underlying concept of *dataflow* that ensures this consistency [7]. In the dataflow software concept, every change of a variable (a cell in the spreadsheet) automatically causes recalculation of all the depending variables (cells). Since this causal relationship is constantly maintained, the reference integrity of data is preserved. In difference to the standard programming paradigm, in which computing actions and outputs are started by user or by scheduled (automatic) events, in dataflow this is triggered for the dependent formula cells implicitly, on every change of any cell value. The new value is stored in the cell, and its content is again constantly visible to the user.

From the data-presentation aspect the dataflow architecture could be depicted as follows.

- The lack of the standard notion of *presentation action (time)*, triggered by user or event, is compensated by:
- The enlargement of the *presentation space* to its extremes (everything is presented all the time).

More elaborate formal analysis of the spreadsheet tables is out of the scope of this paper.

### 2.3 Intermixing the data entry and data calculation columns

From the previous subsection we can deduce that we have basically two types of cells (columns):

- i. Data entry cells (columns);
- ii. Data calculation cells (columns).

Besides the role described by their names, both of these also have the (constant) *data presentation* role. Data entry cells represent the directly entered values, and data calculation cells represent the results of calculations, searches, etc., based on the values of the data and calculation cells on which they depend through their formulas.

In order to make data logically outlaid, the entry and the calculation columns are not radically separated, but

are rather intermixed with each other. As is already mentioned, the redundancy introduced by the calculation columns is a property that is here not considered as anomaly. The only problem introduced by this is how to make the two types of cells clearly distinguishable to the users. Luckily, the modern spreadsheet calculators enable nice formatting options, and also the cell locking mechanism. By using both of them, i.e. by visually differentiating the data entry cells from the data calculation cells, and by locking the calculation ones, we ensure that:

- The user knows where he is expected to enter the data, and where to look for the calculated results;
- The user will be stopped to (accidentally) enter data in the calculation cells and thus spoil the formulas in them.

An enthusiastic data base formalist could be tempted to arrange the tables in such a way that the data entry cells (columns) are always strictly separated from the calculation ones. In this formal separation the true entry values would correspond to the data base attribute values, and the calculated values would be the information excerpted and derived from our database. However, this turns out to be counterproductive and brings no benefit to the tables or the application. In fact, it undermines the very essence of the spreadsheets, which is: *the convenient and logical layout of all data, as we would have them on paper*.

So, the two types of columns should be outlaid in the most appropriate and the most logical way for the problem, or in a way that is most convenient for the user. When planning, the user often wants to have immediate feedback on the input values, which would then help her (or him) to possibly correct the input values, and, thus, through the iteration, do the better planning.

The special features entry cells that will be introduced in 3.2 are a partial exception to the above principle. In the most cases they fit better at the end of the table they correspond to (we place it usually at the right end). Still, since it is just a relative position to other columns of the same table, we cannot consider it as a fundamental separation of the data entry and the data calculation entities.

### 3. PROJECTOR

As hinted in section 1, the Projector's main task is to help in planning and supervising of the labor hours and costs, as well as of all other project's costs. Our project manager is not preoccupied by coordination of the jobs and Gantt charts, but by thinking of how much the jobs cost in the planning phase, and how much they really cost after being completed. By using Projector, the manager wants to inspect the efficiency of different payment models applied to different workers, and to see how they influence the productivity by comparison of the planned and spent hours.

In this section we give a brief presentation of the Projector's capabilities from the user's standpoint.\* The Projector spreadsheet application was designed by having in mind the planning and tracking of projects with the following attributes:

- i. The project starts from some initial state or set of conditions, with possible starting costs (e.g. the costs of acquired assets, bought equipment and similar) which are to be charged to the project.
- ii. The project is defined within an international environment, so the usage of several definable foreign currencies is required. All price inputs must be easily accomplishable in a few user-chosen foreign currencies, and visible simultaneously in the two of them. This is to ensure immediate spotting of the financial indicators on international level.
- iii. The project generally involves heterogeneous labor force, with different competences, work prices, labor taxes and obligations, and requires elaborate tracking of all the indicators regarding the planned and spent labor hours and the corresponding labor costs.
- iv. The project consists of, and can be divided into, a series of jobs (tasks) or the corresponding work terms, for which workers with different skills could be assigned, and which are contracted in different ways.
- v. The project requires some material costs, as well outsourcing costs, which must be included in the total costs surveys.

#### 3.1 Projector's main features

In all the tables elegant entry methods are provided, so that user is never forced to take a side tool to adopt her (or his) data to the required ones.

Among these we outline:

- All the prices (work unit, job price, material and outsourcing price) can be entered in one of the several possible user definable currencies (three in the current implementation, easily expandable to more).
- All the entered cost values *are shown* in the two user definable primary chosen currencies.
- The list of workers (collaborators) is defined with their labor price being connected to the status in the project. To each worker her (or his) labor tax type (percentage) and tax liability can be assigned.
- Each job must be attributed its payment model, and its worker, from the corresponding lists. The payment models are: *paying the agreed fixed price*, *paying the planned hours* (similar to the previous), or *paying the hours spent*.
- The special emphasis is given to the labor hours and costs, which is in the most cases a critical part of every project. Every job has its planned time frame, the explication of the labor costs, planned and spent labor hours, finalization time frame, job stage, and notes on payments.
- Projector leads a project manager from initial planning to its end. It helps through the tracking phase, and records all the workers' earnings and payments.
- Consistent conditional formatting with adjustable parameters is applied to all the relevant cells. They warn about the unfavorable values in alarming colors (red, orange, white), and show favorable or normal values in soothing colors (green, blue).
- Besides locking of all the data calculation cells, the data entry cells are designated by the underlined column titles. Additionally, the data entry cells (col-

---

\* The details on the programming implementation are beyond the volume of this paper.

umns) with the special features (see the next subsection), are always positioned at the right end of the relating table and are border with a double line (confer discussions in 2.3 and 3.2).

### 3.2 Special features data entry cells

If we want the user to be able to enter the labor cost in different currencies, the data entry cells must be organized separately from the data calculation cells showing these values. One of the crucial reasons for this is that the same cell in a table calculator cannot accept both, the values and the formulas. Projector solves this by the help of special data entry cells (columns), usually being placed at the right end of the table. In the case of entering prices, two columns are needed. The first one is for the price numerical value, and the second is for choosing the currency from the list of available options. We call these type of cells *selectable data form entry cells* (columns). On the basis of their values (selections), the prices can be calculated and presented in the separate data calculation cells (columns), placed in the most appropriate position among the other columns in a table.

The example of this is shown in Figure 1 (the table presented here is discussed also in 3.3.1). The last four columns are bordered with the double line to emphasize that these are the data entry columns with special features, including the selectable data form. Of these, the first two are for entering the work hour price. The first one serves for entering the price value, and the second one for the currency. The last two columns define the labor taxes, but just for inspecting their influence on the labor prices. The taxes are attributed directly to the workers (see 3.3.2 and Figure 2).

Here we see how the introduction of the separate data entry columns—needed to accomplish the selectable data form—required the addition of the calculation columns, 3 – 4 and 6 – 7 within the table.

The introduced redundancy is fully justified, not only by the added functionality, but also because of the good table layout, simpler further calculations, and preserved data consistency, because the user cannot input inconsistent price values in different currencies.

Another special feature added here is the *Common Choice (Cm.Cho.)* option, which can be selected from the drop-down list. The last three columns in Figure 1 have this functionality. By selecting the Cm.Cho. option for a particular cell within a column, the cell will be assigned the same value as in the column title cell with the name (or name appendix) *Cm.Cho.* By selecting the Common Choice option as a default value for all the column cells, they will be all set to that same value. This is very useful when most of the cells have the same, “common column” value. If some cells are different, we change them to the desired different values. The global change of all the cells set to the Common Choice option can be easily done by changing the single value in the top title cell.

In the Figure 1 example, in the second column to the right from the double line, USD currency is selected to be the common choice. This value is assigned to all the cells having the Cm.Cho. value, which are the first three in our case. The fourth cell is set to EUR, and the fifth to GBP. If, for example, we want to change the Common Choice to EUR, it is done by a single change of the title cell to the value of EUR.

### 3.3 Working with Projector

The following directions will help the reader and Projector user to understand its logical structure. In the present version, Projector consists of 6 relationally connected spreadsheets. Their numbers are ordered in the way of suggested data input and generality, from common to special.

Labor Cost Per Hour		Excluding Tax		Tax	Including Tax		Lbr.Cst.	Currncy/ Cm.Cho.	Lbr.Tax/ Cm.Cho.	Tx.Incl. in In.Val.
Description	Abr.	1 hr / EUR	1 hr / USD	Amount	1 hr / EUR	1 hr / USD	?Value	USD	US_LTX	Incl. ?
Audio Tech. Apprentice	AT1	4.12	5.77	30.0%	5.36	7.50	7.50	Cm.Cho.	Cm.Cho.	Cm.Cho.
Audio Technician	AT2	5.49	7.69	30.0%	7.14	10.00	10.00	Cm.Cho.	Cm.Cho.	Cm.Cho.
Audio Supervisor Tech.	AT3	6.87	9.62	30.0%	8.93	12.50	12.50	Cm.Cho.	Cm.Cho.	Cm.Cho.
Audio Supervisor & Tutor	AT4	11.11	15.56	35.0%	15.00	21.00	15.00	EUR	DE_LTX	Cm.Cho.
Aud. Engineer & Designer	AT5	17.14	24.00	33.3%	22.85	31.99	15.00	GBP	UK_LTX	Not Incl.

**Figure 1. Labor Cost Per Hour table on the sheet S1. General & Summary.** The figure illustrates the table with the data entry columns separated from the main presentation table. These are the four columns bordered with the double line, which include the special input features. The first two of them enable the selectable data form input, by entering the labor price value and the currency. Where the Cm.Cho. (common choice) option is selected for the currency, the cell is set to the same value as in the light shadowed title cell named *Currncy/Cm.Cho.* In our example this cell value is USD = U.S. Dollar. The first three rows have the same currency (Cm.Cho. = USD), and the last two are different (EUR and GBP). This mechanism saves the entry time when most of the cells have the same attribute values. It is applied in the same manner to the last two columns for the selection of the labor taxes (the first three cells are U.S. Labor Tax, and the last two are German and UK Labor Tax), and the specifications if the labor tax is included in the entered price or not (included, except for the last one). The taxes here are only for the purpose of defining the entered values, and showing their influence on the labor price (confer also Figure 2). Besides the last four, the first two columns are also data entry columns (indicated by the underlined titles), but without the special features. All the others are data calculation columns (confer 2.3). [Conversion rates on the date of writing the article were approx. 1EUR = 1.400 USD, 1GBP = 1.600. The labor tax ratios are approximate or fictional].



### 3.3.1 Sheet #1: General & Summary

In the first sheet, the general data about the project, as well as the general data defining the business environment are defined.

**General project data.** At the top left of the first sheet there are a few tables for defining the project name, ownership, general description, technical details and initial investments. E.g. if we are into constructing a new motor with inner combustion on hydrogen derived directly from the water reservoir, the initial investment can be the price paid for the old machine that we shall modify. If we are into improving a piece of software bought from another source, this could be the price of a licensed version of some programming tool paid earlier, but charged to this project. Some of the title cells can be modified according to the user's needs and the project's specifics.

**General financial data.** Next to the project-related general data, at the top right of the sheet #1 are several helping tables that define the project parameters. The user can define her (or his) two currencies of choice, in which all the costs and price values will be constantly presented for immediate reference and comparison (confer Figure 1 and the next paragraph). One more currency is provided for the cost inputs in the data entry cells (see def. in 2.3).

For the total of  $n$  different currencies the  $n \times n$  conversion rate matrix is defined. In the matrix, there are  $\binom{n}{2} = n(n-1)/2$  independent currency conversion rates, the same number of dependent (reciprocal) conversion rates, and  $n$  identities (stating that a currency is equal to itself), leading to the total of  $n \times n$  values. In our case  $= 3$ , but the number of the data entry currencies can be easily enlarged according to the needs.

There are a few more tables defining the usual labor related constants, like the number of work hours in a

work day and a week. The choice of several labor taxes from a user-editable and expandable table is provided.

**Labor Cost Per Hour.** This table was already discussed in 3.2, where we have stressed that the last four columns are the (separated) data entry columns. Besides these, the first two columns are also data entry columns (indicated by the underlined column titles). In them, the qualifications or possible worker's position in the project can be defined, together with the corresponding abbreviation which serves as a key. The columns 3 to 7 are all calculation columns of obvious meanings.

**Other parameters and names.** Several other helping tables, like the tables of the possible payment models, job stages, and others, ensure the full control over the behavior of the table calculations, extraction of data, numerical and textual constants, and similar parameters.

All the names that will appear in the multiple choice lists are consistently written in either the columns or rows of appropriately constructed tables, so that they can be easily checked and changed.

### 3.3.2 Sheet #2: Workers & Labor

The second spreadsheet and all the following ones start with the locked copies (always consistent with their originals) of the summarizing planned data from the sheet #1. This serves as a short reminder of the most important facts when planning and running the project's concrete stages, like the labor force in this case.

**Labor force.** The sheet #2 main table is called *Labor Force* (Figure 2). The worker's names and short ID's are entered here. The ID serves as the key for searches throughout the data base. Also, the worker's position in the project is chosen here (defined in the Labor Cost Per Hour table above), together with the labor tax value and tax liability.

Labor Force												
Worker				Net Price / hr (No Labor Tax)		Labor Tax			Tot. Price / hr (Wth. Lbr. Tax)		Entry Cells	
#	Name	Wrkr. ID	Qualif. in Proj.								Cm.Cho.	Liability Cm.Cho.
				/ EUR	/ USD	Type / Cntry.	Amount	Liability	/ EUR	/ USD		
1	Alliston, Alice	AA	AT1	4.12	5.77	US_LTX	30.0%	Worker	5.36	7.50	Cm.Cho.	Cm.Cho.
2	Brown, Bob	BB	AT2	5.49	7.69	US_LTX	30.0%	Worker	7.14	10.00	Cm.Cho.	Cm.Cho.
3	Cooper, Cindy	CC1	AT2	5.49	7.69	UK_LTX	33.3%	Worker	7.32	10.25	UK_LTX	Cm.Cho.
4	Havaš, Ladislav	LH	AT4	11.11	15.56	HR_PRD	50.0%	Project	16.67	23.33	HR_PRD	Project
5	Logožar, Robert	RL	AT5	17.14	24.00	HR_PRD	50.0%	Project	25.71	36.00	HR_PRD	Project
Average :				8.67	12.14	38.7%			12.44	17.42		

**Figure 2. The sheet #2 main table: *Labor Force*.** The first column automatically counts the workers' names entered in the second column. The third and fourth columns are for the entry of the worker's IDs, and the qualification (position in the project). These 3 columns are the standard data entry columns with the underlined titles. The last two entry columns (with the emphasized double line frame) have the special input features, in this case the option lists that include the Cm.Cho. (compare also 3.2 and Figure 1). Here the labor tax type can be chosen for each worker, including the tax liability. For Alice and Bob the U.S. Labor Tax is chosen through the Cm.Cho. option. The other selections are obvious. All the other columns, excluding 1, that is: 5 – 11, are calculation columns that use values from the input data columns and the tables defined on the sheet #1. The net price is taken from the Labor Cost Per Hour table (Figure 1), and the tax amount from the table of taxes.

By providing that this data are separately definable for each worker, we enable engagement of collaborators from different countries, and from both, inside and outside the project owning company.

To the right from the Labor Force table, several presentation tables expand, all having coincident worker's rows. Such alignment enables simple adding or deleting of the workers, by using the row copy and insert, or row delete functions, provided in MS© Excel spreadsheets.

All 11 tables from the sheet #2 which are related to the labor force and its different aspects, have unique numbers of the form: 2.1.n, with  $n = 1, 2, \dots, 11$ . They are:

1. Labor Force (worker name, ID, labor tax type and liability, net and total labor price per hour, Figure 2);
2. Work Planned (worker's labor hours systemized per payment models, also in cash price);
3. Work Completed (as above for completed work);

4. Work Not Completed;
5. Work Total (both, completed and not completed work);
6. Bonus or Penalty;
7. Worker's Earnings;
8. Payments paid to Workers;
9. Labor Tax Payments;
10. Payment Dates and Cumulative Amounts (1);
11. Payment Dates and Cumulative Amounts (2);

Tables 1, 6 and 9 have data entry columns. All the other tables have only the data calculation columns, and serve solely for the data and information presentation purposes.

The last two tables serve also as helping tables, with intermediary results needed for finding out which jobs can be considered as paid, taking into account the payments paid to each worker (see 3.3.3 and Figure 3).

Jobs (Wrk.Trms.) Specifications												
Job or Work Term				Job Price Calculation?					PrcPerJbTyp ?		JPrC: AcctntTot	
No.	Description	Job Pymn. Model	Done by Wrkr. ID	Dates Planned		Net Hours (Without Breaks)			Net Hours + Corrections		Job Price Selected	
				First	Last	Plnnd. (Spent)	Spent (Plnnd.)	Crctns. Apprvd.	Plnnd. (Spent)	Spent (Plnnd.)	/ EUR	/ USD
1	Inspection of the initial state and the problem to be solved.	PJFxPr	RL	01.05.11	05.05.11	1.39	4.00		1.39	4.00	35.71	50.00
2	Investigation of the similar problems, and reporting on them.	PrJbPI	AA	02.05.11	06.05.11	10.00	4.00		10.00	4.00	53.57	75.00
3	Designing the prototype model.	PJFxPr	CC1	07.05.11	12.05.11	19.50	4.00		19.50	4.00	142.86	200.00
4	Building the prototype model.	PrHrSp	LH	12.05.11	20.05.11	24.50	24.50		24.50	24.50	408.33	571.67
5	Testing the model and reporting to the team.	PJFxPr	RL	20.05.11	21.05.11	2.22	4.00		2.22	4.00	57.14	80.00

Figure 3a. The sheet #3 main table: Jobs (Work Terms) Specifications, the left part. The first part of the table (that could fit to the paper width) has the data entry columns 2 – 6: Description of the job, Payment Model, Worker ID, and Dates Planned. Values in col. 3 and 4 are to be picked up from the drop-down list. The data calculation columns are 1, and 7 – 13. The column 1 automatically counts the jobs with description that is not empty. The net hours and correction hours (columns 7 – 11) are calculated from the separated data entry columns (Figure 3b to the right) according to the job payment model. The same is valid for the job price in the columns 12 – 13. The columns that need not be presented can be hidden by the user.

← Job Price Type Slctd.								
Job Prc.Cmplt. Cumltv.Pr. Wrkr.		<u>Dates Realized</u>		Job Stage / Complt.	Jb.Dsc. Exst. & Crct.	P a i d		Comments, Objections, <u>Reclamations</u>
/EUR	/USD	<u>First</u>	<u>Last</u>			% of Jb.	Date	
35.71	50.00	04.05.11	07.05.11	✓	✓			Comments about the first job.
53.57	75.00	04.05.11	10.05.11	✓	✓	100.0%	20.05.11	
142.86	200.00	07.05.11	14.05.11	In Proc. as Plnnd.	✓	75.0%	20.05.11	
408.33	571.67	14.05.11	22.05.11	✓	✓	100.0%	22.05.11	
92.86	130.00	22.05.11	25.05.11	In Proc. wth Dlvs.	✓			

Job Price & Hours Input Table					
Job Price			Time / hr		
Input Value	Currency (Cm.Cho.)	Lbr. Tax in In.Val.	Plnnd.	Spent	Correctns. / hr
	USD	Incl. ✓	In.Val.	In.Val.	In.Val.
1	50.00	Cm.Cho.	5.00	4.00	
2		Cm.Cho.	10.00	4.00	
3	200.00	Cm.Cho.		4.00	
4		EUR		24.50	
5	50.00	GBP		4.00	

Figure 3b. Sheet #3 main table: Jobs (Work Terms) Specifications, the right part, and the neighboring: Job Price & Hours Input Table. The right part of the main table has columns 14 – 22. The data entry columns (with the underlined titles) are 16 – 17, 18, and 22, with the meaning obvious from their titles. The calculation columns 14 – 15 are the helping columns for finding out which jobs are covered by the payments (see Figure 4). The Job Stage/Completeness column has 9 values describing the possible job stages. The last stage, denoted by the check mark, is a successful job completion. The column 19 registers if the job is correctly described. If not, it is not accounted in the searches. The next columns, 20 – 21, show the percentage of the job covered by the payments paid to the worker up to now, and the date of the earliest payment covering that job. Next to the main table is the corresponding input table which has selectable data form entry columns for the job price in different currencies. These cells, if entered, will explicitly define the job price. If not entered, the job price will be calculated in the helping calculating table to the right (not shown) on the basis of the entered labor hours. Next to the job price are the job hours' entry columns for entering the planned and spent hours, and their possible afterwards corrections.

### 3.3.3 Sheet #3: Jobs (Work Terms)

This spreadsheet is central for the planning and tracking of the project's jobs, with the main table named **Jobs (Work Terms) Specifications**. There are two associate tables: **Job Price and Hours Input Table**, and **Job Price and Hours Calculation Table**. The first of these tables, and the left part of the second one, are shown on the Figure 3a and 3b on the previous page.

**Jobs (Work Terms) Specifications.** This was the Projector's starting table. It follows the simple logic of dividing a project into jobs or work terms, and of assigning an available worker to every job. If two workers should be doing the "same job", we subdivide it in two and enter them in the two rows adequately described.

No analysis of the jobs' mutual dependence is done here – Projector leaves this to humans. But it will calculate the job prices, record planned and spent hours, keep the track of the planned and realized time frame, present the short description of the job stage, and finally, provide the insight if the job is paid or not, based on the payments in the sheet #4 (see below). Some more details are given in the figure captions.

### 3.3.4 Sheet #4: Labor Payments

This spreadsheet consists of one main table: **Labor Payments** (Figure 4a and 4b).

**Labor Payments.** This table is a typical representative of applying the concepts described in section 2. The data entry and calculation columns are intermixed with each other to provide a logical layout of the table.

To correctly collect the worker's earnings, to enable multiple payments paid to the same worker, and to keep the record of all the payments correctly, this table relies on the helping tables 10 and 11 described in 3.3.2. The little left-pointing arrows next to the rightmost column (Fig 4.b) help in tracking of the last payment paid to the worker.

### 3.3.5 Sheets #5 and #6: Parts & Materials, Outsourcing

The sheets #5 and #6 are constructed for tracking of the parts and materials, and the costs of outsourcing. The concepts and solutions described for the previous spreadsheets are consistently followed, enabling the user to input data easily, intuitively and creatively. More in detail description of these tables is left for the technical documentation.

### 3.3.6 From the general view to the details and back

Figure 5 summarizes the Projector's spreadsheets, and suggests possible data entering paths. Filling out the sheets in the order of their numbers would result in a top-down project design method. But the user can start from anywhere, defining the missing data later on. E.g. the planning can start from the sheet #3 main table by dividing the project into smaller jobs. The jobs will not be checked by the program as "correctly and completely described" if not having properly assigned workers (confer 3.3.3 and Figure 3), but the project manager can do this first and fill in the missing data later.

Labor Payments													
To Worker				Earnings: E1 + E2 →				To P a y		Earnings: E1 + E2 + E3 →			
Py m. #	N a m e	Wrkr. ID	Qualif. in Proj.	Amount		Dates		Amount		Amount		Dates	
				/ EUR	/ USD	Earlier E1 – E2	Pymnt. Due	/ EUR	/ USD	/ EUR	/ USD	Earlier E1 – E3	Pymnt. Due
1	Alliston, Alice	AA	AT1	53.57	75.00	10.05.11	25.05.11	0.00	0.00	53.57	75.00	10.05.11	25.05.11
2	Cooper, Cindy	CC1	AT2	0.00	0.00	–	–	–	–	142.86	200.00	07.05.11	22.05.11
3	Havaš, Ladislav	LH	AT4	408.33	571.67	22.05.11	06.06.11	0.00	0.00	408.33	571.67	22.05.11	06.06.11
4	Logožar, Robert	RL	AT5	35.71	50.00	07.05.11	22.05.11	35.71	50.00	92.86	130.00	07.05.11	22.05.11

**Figure 4a. The sheet #4: Labor Payments, the left part.** The left part of the table has just one data entry column (number 3) for the worker's ID. The payment is fully defined by the workers' ID and the entrance of the amount and the date of payment (there can be more than one payment to a worker). In order that the table be useful for the manager, it shows the worker's earnings, E1 + E2 being the sum of earnings for the jobs completed (E1) and the bonus earnings (E2). The E3 presents the uncompleted jobs, which can be also considered for paying, in advance.

**Figure 4b. The sheet #4: Labor Payments, the right part.** The right part of table shows the remaining two columns for the sum to be paid on the E1 + E2 + E3 scheme basis (see Figure 4a). The next three columns to the right (columns 17 – 19) show the total payment given to the worker and the last payment date. As usual, the special data entry columns at the right end of the table have the double-line border. Again, the usual convenient features for the data entry are provided, including a few options for setting the payment date (e.g. to be equal to the job completion date).

To Pay		Payment			Payment Entry			
Amount		Amount		Date	Pymnt. Amount	Curren- cy/ (Cm.Cho.)	Date Entry	Date Ch/ (Cm.Cho.)
/ EUR	/ USD	/ EUR	/ USD					
0.00	0.00	53.57	75.00	20.05.11	75.00	USD	20.05.11	Cm.Cho. <
35.71	50.00	107.14	150.00	20.05.11	150.00	USD	20.05.11	Cm.Cho. <
0.00	0.00	408.33	571.66	22.05.11	408.33	EUR	22.05.11	CmplDt. <
92.86	130.00			-		EUR		Cm.Cho. <

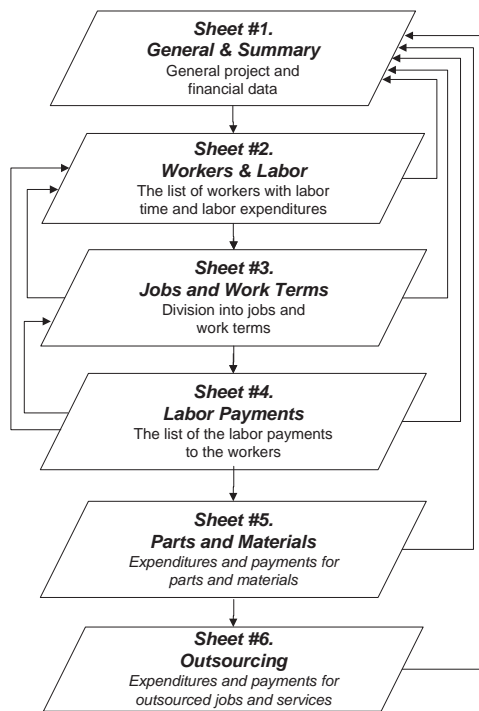


Figure 5. The Layout of the Projector's Spreadsheets.

After having defined precisely what has to be done, she (or he) goes back to the Sheet #2 and chooses a suitable team of workers. So, one can fully follow her (or his) intuition and combine different design approaches as desired: top-down, bottom-up, or intermediary. In any time the user can go from one sheet to another, checking out some other aspects of the project, or filling in some new data.

#### 4. CONCLUSION

Programming is an endeavor that starts with our wish to get better insight and better control of systems and processes that can be very complex. In this paper we have illustrated how such a (naïve) wish to plan and track the labor costs of a project by a "simple Excel table", led us to development of an elaborate spreadsheet application in MS® Excel.

Being driven by our own custom demands, we took a different approach to project planning and tracking, putting the emphasis on the quantitative indicators of labor hours and costs, as well as on other project expenditures. In today's world, it is the planning and tracking of each worker's and coworker's tasks, duties, planned and spent time, planned and finalization terms, that is crucial for having a good insight and foresight in both, the projected and the final price of any project.

The simple but efficient programming environment of the table calculator proved to be sufficient and suitable for the task, and creation of the Projector application.

In the introductory sections, we have discussed a few

specifics of the spreadsheets based on the dataflow concept. We have briefly and informally contrasted the ways of organizing data in the spreadsheet tables to those in the databases. In the table calculators we mix the data entry cells (columns) with the data calculation cells (columns). The latter are derived from the original input data and the project parameters on the basis of the cell's formula. They can serve for the presentation of new information, or as helping variables for further computation and data extraction.

In section 3 we have given an outline of the Projector capabilities. Some of the specifics are the special features data entry cells. In them we have implemented the selectable data form entry cells and mechanisms for simplifying the input in multiple cells having equal attribute values (Common Choice option). These solutions are commented and presented in several example tables. We have also taken a quick tour through the Projector's spreadsheets, following the top-down project planning. After defining the initial parameters and the labor force, by defining jobs or work terms, project manager can have immediate insight in the labor costs, and also track down the payments paid to the workers.

There are many specialized and detailed calculation (presentation) tables in Projector. They offer insight in different statistical aspects of the entered data, on general and particular level. E.g. we can track the general job completion, as well as the job completion as done by particular workers. We can track the worker's earnings and the total labor costs, and all these can be viewed from several different aspects. The many functional details, however, deserve a separate discussion. The same is true for many interesting, and sometimes very complex, data extraction and data analysis solutions in the spreadsheet calculators. All this, together with many further improvements and upgrades that could be added to Projector, are the good hints for a future work. Hopefully we have provided a solid foundation for that.

#### 5. REFERENCES

1. Microsoft® Project, <http://office.microsoft.com/en-us/project-help/>.
2. Wallace Clark and Henry Gantt, The Gantt chart, a working tool of management. Ronald Press, New York, USA, 1922.
3. Peter W. G. Morris, The Management of Projects, Thomas Telford, 1994.
4. Free Project Management Software, <http://www.softwareprojects.org/free-project-management-software.htm>.
5. Walkenbach, J. „Excel 2007 Bible“, Wiley Publishing, Inc., Indianapolis, Indiana, 2007.
6. Codd, E. F., A Relational Model of Data for Large Shared Data Banks, Communications of the ACM 13(6): 377-387, 1970.
7. W.R. Sutherland, The On-line Graphical Specification of Computer Procedures, MIT 1966.